Feature

Visual Servoing for Online Facilities

Visual servoing for online facilities aims at automated manipulation of instrument control parameters based on visual-scene interpretation. It provides a layer of computing that hides the latency in the Internet environment and simplifies the use of scientific imaging instruments. This approach makes remote real-time imaging experiments feasible over unpredictable wide area networks.

Bahram Parvin

John R. Taylor

Daniel E. Callahan

William E. Johnston

Ulrich Dahmen

Lawrence Berkeley National Laboratory

Distributed multimedia systems for remote instrumentation aim to provide access to facilities for collaborative research.[1-3] The enabling technologies for these systems include a human-machine interface, shared visualization environment, and the required communication infrastructure. The impact of such capabilities is twofold: They extend and enhance the use of scarce resources and provide avenues for closer collaboration.

In the research environment, the natural evolution for such systems is to provide the computational framework to analyze images, extract meaningful information from a video sequence in real time, and manipulate experiments, over a wide area network. Such a framework serves as a broad base for general laboratory automation.

Our efforts focus on developing a framework for remotely manipulating microscopic objects. Two diverse applications demonstrate our framework's usefulness: microdissection of DNA molecules and in-situ examination of crystal formation. Microdissection and subsequent amplification of DNA molecules allow for rapid closure of gaps in the genomic library. In-situ experiments reveal information about the thermal and morphological properties of crystal structures. For more information see the "Genes and Crystals" sidebar on page 58.

Both experiments are labor-intensive and can benefit from automation. Their common thread in our framework is a set of visual routines, which we implement over a distributed client-server software architecture for better throughput, scalability, and modularity. This model also enables remote operation of the proposed experiments over a wide area network, which is particularly significant for dynamic studies. Previously, thermal drift, rapid topological changes on the specimen, and unpredictable WAN latency made this class of experiments unsuitable for remote operation. We have demonstrated that intelligent visual interpretation and the use of this information for control will compensate for the Internet environment's latency and ease the use of a complicated and centralized instrument.

The testbeds for these experiments are a florescence optical microscope (for DNA microdissection) and a 1.5-MeV transmission electron microscope (to study crystal formation) that is operated by the National Center for Electron Microscopy. The experiments are being performed as a part of the Department of Energy's DOE 2000 National Collaboratory Program.

APPROACH

Our general approach is to separate interactions into two groups: those that don't require low-latency communication and those that do. Basic human interactions that establish control system parameters, such as gross positioning and identifying objects of interest, do not require low-latency communication. Tasks such as autofocusing and thermal-drift correction do. In this framework, visual routines provide the necessary information to drive the control system, hiding and automating tedious, remote operations. It also makes remote, real-time experiments feasible through visual servoing over a high-speed, local area network. Figure 1 illustrates this approach.

COMPUTATIONAL ENVIRONMENT

The computational environment that implements automated control in the local environment must acquire images, process them at the required rate, and manipulate several functions that operate the microscopes. Our strategy for partitioning these operations is based on design philosophy (which emphasizes scalability, modularity, reusability, and cost) and on the data acquisition components available for our various hardware platforms. For these reasons, we used two Sun Microsystems Sparc 20s

0018-9162/97/$10.00 © 1997 IEEE

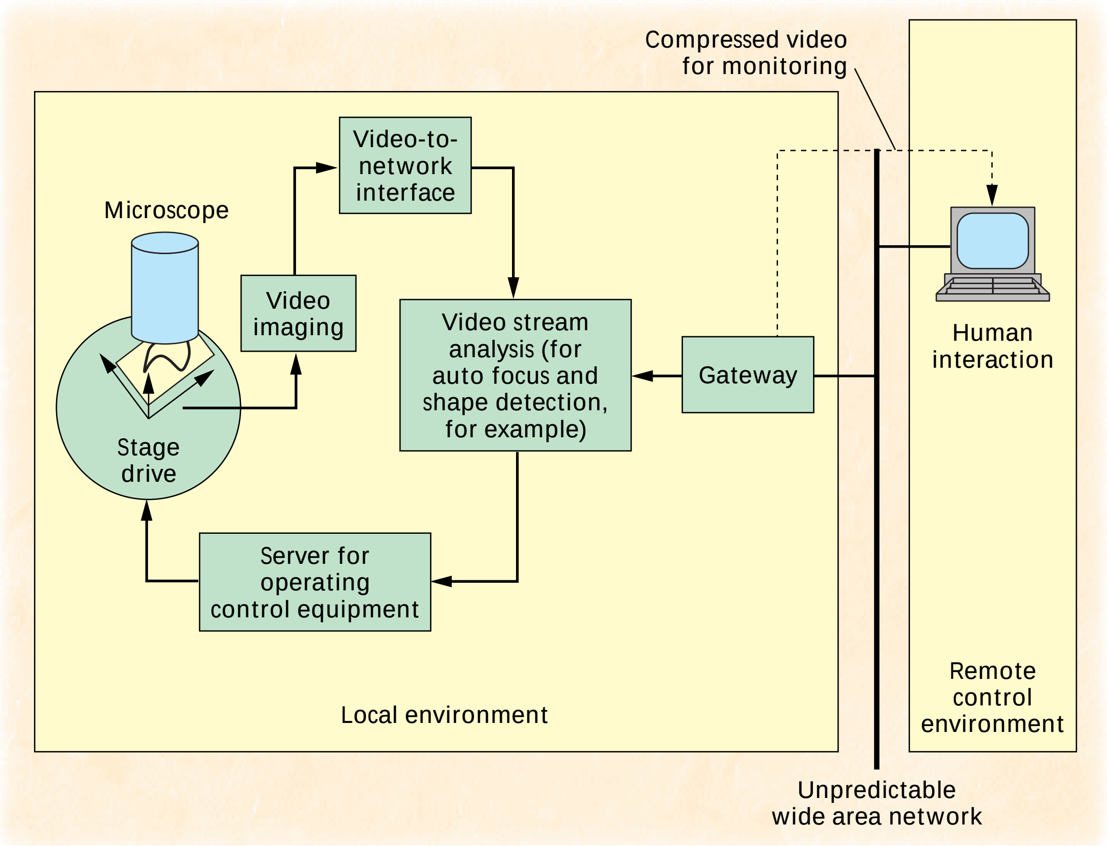
Compressed video
for monitoring
Microscope
Video-to-network interface
Video imaging
Video stream analysis (for auto focus and shape detection, for example)
Gateway
Human interaction
Stage drive
Server for operating control equipment
Local environment
Remote control environment
Unpredictable wide area network

Figure 1. Remote operation architecture.

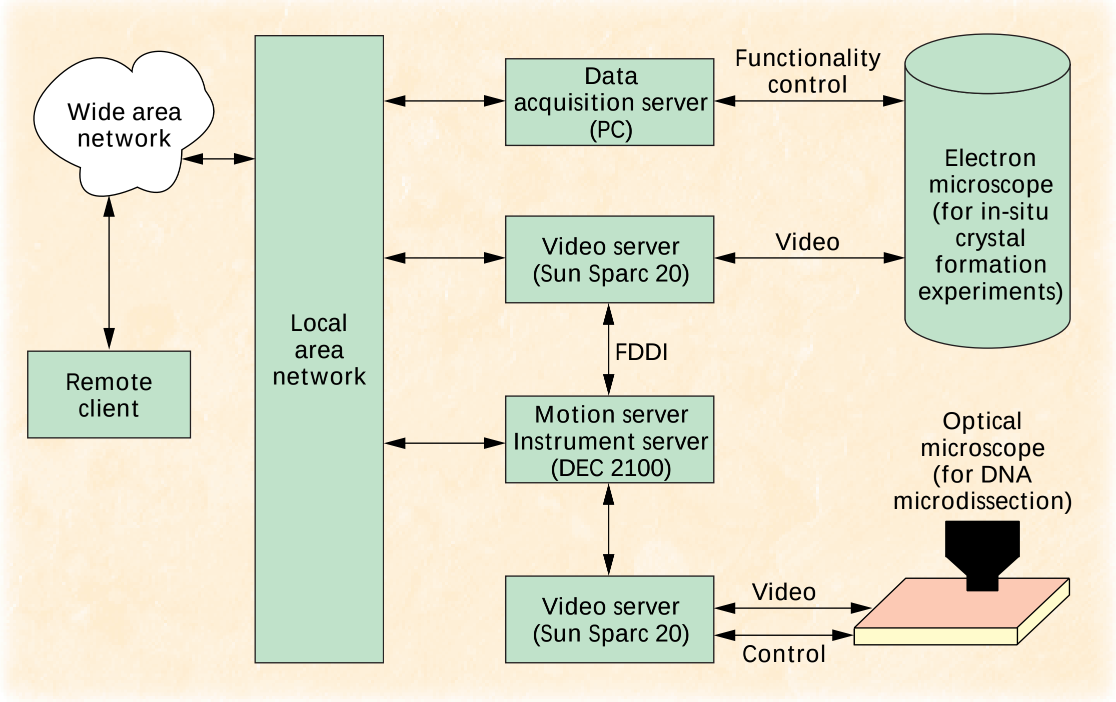
Wide area network
Local area network
Remote client
Data acquisition server (PC)
Functionality control
Electron microscope (for in-situ crystal formation experiments)
Video server (Sun Sparc 20)
Video
FDDI
Motion server Instrument server (DEC 2100)
Optical microscope (for DNA microdissection)
Video server (Sun Sparc 20)
Video
Control

Figure 2. Computational environment for remote, in-situ microscopy.

for image capture (one for each microscope), a Digital Equipment Corp. 2100 symmetric multiprocessor for CPU-intensive operations, and a PC for data acquisition. Physically, the Sparc 20s and the PC operate near the microscopes, while the DEC 2100 is in another laboratory building and connects via a LAN. The Sparc 20s and the DEC 2100 transfer images over a fiber distributed data interface (FDDI) ring at 100 Mbytes per second, as Figure 2 illustrates.

VISUAL ROUTINES

In remote microscopy, the system must provide the

Genes and Crystals

Two applications drove our development efforts.

Microdissection of DNA Molecules

Precise, automated microdissection of stretched and immobilized single DNA molecules has several important applications. If multiple (20 to 200) samples can be recovered from the same region of identical molecules that have been stretched to the same extent, it is possible to amplify this material using the polymerase chain reaction. Scientists can then use microdissected and amplified DNA as a probe to identify members of large genomic libraries that contain DNA identical to that of the dissected region.

Human genomic libraries are large arrays of clones (yeast or bacterial cells) in which each clone contains one specific fragment of the human genome. The human genome consists of approximately 4 billion nucleic acid bases from 24 different chromosomes, and each human chromosome contains a single strand of DNA that is 50 to 250 megabases. This genomic DNA is broken into fragments, and each clone of a genomic library contains a single fragment. Yeast clones containing yeast artificial chromosomes may contain intact human DNA fragments as large as one megabase, whereas other types of more stable bacterial clones contain smaller intact fragments (75 to 300 kilobases).

In all cases, the clones in genomic libraries contain random and different but sometimes overlapping DNA fragments. The members of the library must be identified in a way that reconstructs the original order of the fragments in the human chromosomes. This process is first performed using genomic libraries containing clones with very large fragments (yeast artificial chromosomes). Next, members of this ordered library serve as templates on which to order clones from other libraries in which the members contain smaller DNA fragments. One of the Human Genome Project's goals is to obtain ordered sets of overlapping clones containing DNA fragments short enough (500 to 2,000 base pairs) to submit to automated DNA sequencing machines. These ordered sets of clones can then be propagated and shared between laboratories for gene discovery and medical research.

The proposed application is designed to target regions on large-fragment clones where there are gaps in previously ordered sets of P1 (bacterial) clones. Closing gaps is currently the most time-consuming process in obtaining ordered sets of clones. Scientists have proposed that small, specific regions (5 to 10 kilobases) of the gap be recovered, amplified, and then used as probes to identify clones that fill in the missing regions. The target region is specified in terms of fractional length. Since an extended yeast artificial chromosome DNA can extend beyond one field of view, it would be difficult for a human operator to identify and dissect the correct region along a featureless DNA molecule. Any attempt to chemically process the DNA so that a human operator can visualize the target region is likely to damage the DNA and make it impossible to amplify. In addition, for successful amplification, microdissections should be performed quickly (to avoid excessive illumination), precisely, and repetitively (20 to 200 times on separate molecules). Thus, automating the process is essential.

From an algorithmic perspective, the system needs to detect the position of a microcapillary over the coverslip, select a DNA molecule, designate a position on the molecule, scrape a piece of that molecule, and verify the scraped area along the molecule.

Dynamic In-Situ Microscopy

This class of scientific experiments study a specimen perturbed by an external stimulus. The stimulus can be a variation in temperature, electromagnetic field, or the environment's chemical or biological composition. Interaction of external stimuli with the specimen can result in sample drift, shape deformation, changes in focus, and other responses.

The application we studied examines crystal shape changes (between liquid and solid states) as a function of thermal cycles. This is the *shape-equilibrium problem*, in which we optimize the thermal path to produce identical shapes as the crystal cycles between solid and liquid states. During in-situ experiments, the operator constantly adjusts the instrument to maintain focus and compensate for various drifts. This labor-intensive task requires a high-bandwidth video link and is nearly impossible to perform remotely because of network latency. Therefore, to accomplish remote operation, we must resort to automated video analysis tools.

look and feel available to the local operator and hide the inherent latency in the WAN. Look and feel is achieved through an appropriate user interface. Hiding the network latency is achieved through visual servoing.

From an algorithmic perspective, a common set of requirements for our two applications includes

- image compression and autofocusing,
- self-calibration,
- object detection,
- tracking, and
- servo-loop control.

These requirements are realized through a set of computational techniques that have been developed in our laboratory[4-6] as well as by modification and extension of existing approaches.[7,8] We implement each requirement in a separate computational component.

Image compression and autofocusing

Both image compression and autofocusing use the wavelet transform as their underlying principle. We use Ingrid Daubechies's kernels,[9] which are simple, orthogonal, highly localized, and separable for two-dimensional processing. The wavelet transform's main advantage is that it can represent local feature activities at multiple scales through spatial decimation. During image compression, our system ignores low-order wavelet coefficients and encodes the remaining ones in 16 × 16 pixel blocks. The remote user has full control over what percentage of wavelet coefficients the system uses for compression.

Autofocusing has two operational modes: initialization and runtime. The difference between these modes is based on the scope of the search for the best focal position. Our system measures focus quality by accumulating the square of the wavelet coefficients in a particular subband. At runtime (as a user heats the specimen), the system automatically makes small focal position adjustments to compensate for three-dimensional changes in the shape of inclusions (crystals in any form). Similarly, for DNA microdissection, the coverslip (a thin glass holding a specimen on a microscope slide) is not perfectly flat. Our system slightly adjusts the focal position as the stage that holds the coverslip moves from one location to the next.

Self-calibration

The motion of the *xy* stage (a platform holding the specimens in both applications) or the robotic arm (for DNA microdissection only) is precisely mapped into the actual pixel size. This mapping is an affine transform estimated by using either low- or high-level features. Because most microscopy applications have a narrow depth of field, we constrained the motion to two dimensions to simplify calibration.

In the case of microdissection of DNA molecules, it is also necessary to calibrate the robotic arm. The transform is obtained by projecting a deformable contour model on a microcapillary mounted on the arm. The arm is then moved into two known positions, and the microcapillary's position is reacquired with the deformable model. Calibration of the *xy* stage—for either optical or electron microscope—is performed with the optical flow field.[7] The image motion, as perceived from the viewpoint of the *xy* translation stage, is rotated and slightly off the scale in one axis. A pyramid implementation of the optical flow field is used to rapidly obtain the affine parameters.

Object detection

The system can detect tubular, convex, and circular objects. We model DNA molecules, observed under the optical microscope, as tubular objects; inclusions, observed under an electron microscope, as convex objects; and living cells, observed under the optical microscope, as circular objects.

Object detection is based on perceptual grouping principles. We define grouping as aggregating features—such as line segments or local symmetries—so that certain rules and geometric constraints are satisfied. This is initiated by using Canny's edge detector, and the final grouping is expressed as bounding polygons around objects of interest. This description is then refined, projected, and tracked in subsequent frames using the tracking subsystem. In this context, object detection occurs only in the first frame for initialization. To detect DNA molecules, the object detection subsystem groups U shapes and antiparallel segments (two parallel line segments with opposing directions). In a sense, the subsystem performs grouping along the object's axis of symmetry.

To detect inclusions, the subsystem accumulates line segments as long as they satisfy the convexity constraint. The actual search process is based either on dynamic programming or greedy search; earlier publications explain the details.[5,6] Figure 3 shows examples of DNA molecule and crystal inclusion detection.

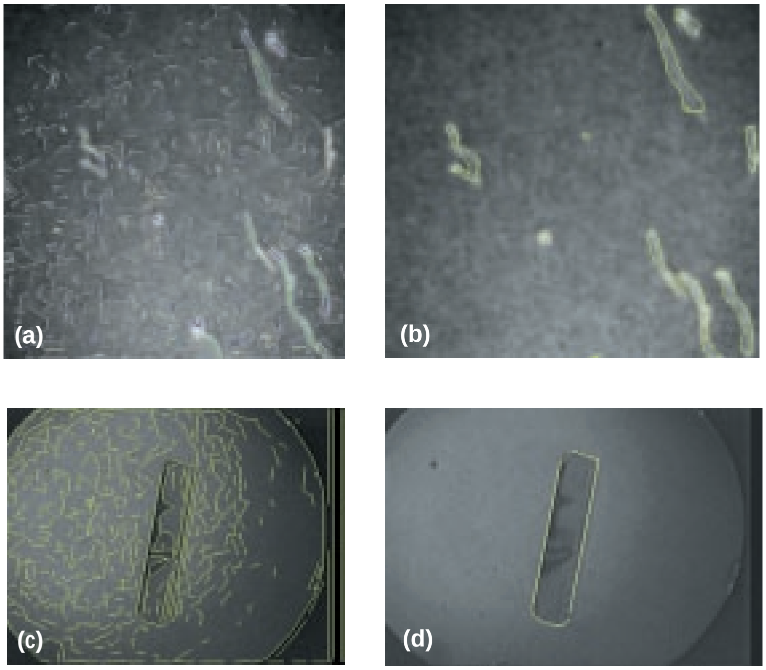

Figure 3. Detection of DNA molecules: (a) original image with Canny's edges and (b) detected tubular objects. Detection of germanium inclusion: (c) original image with Canny's edges and (d) detected convex shapes.

Object tracking

Our system supports two modes of tracking based on high- and low-level feature activities corresponding to deformable contours and optical flow field.

During dynamic, in-situ experiments on the electron microscope, it is necessary to compensate for thermal drift as the specimen is heated. We have built a closed-loop system using optical flow field computation to stabilize the stage where the specimen is located. Current throughput is 4 Hz using a multigrid implementation of the optical flow technique. The actual software implementation uses Posix threads to realize the necessary parallelism under the DEC Unix operating system.

The second mode of tracking is based on high-level shape changes, where a variant of the deformable model[5] is used to quantify and track objects of interest. This approach initially refines the bounding polygon obtained during the detection process and then projects it on each consecutive frame for continuous updating. Conceptually, each point on the refined contour should have high gradient and good directional continuity with both the low- and high-level features. Low-level features refer to the edge direction among nearby edges, and high-level features refer to the direction of a nearby line segment corre-

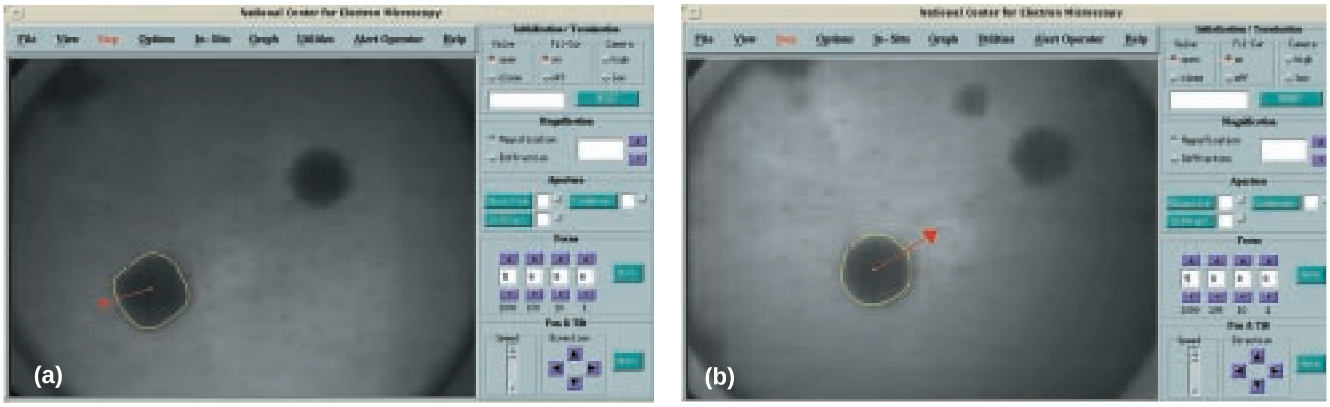

Figure 4. Tracking and compensating for drift during heating and cooling cycles. The direction of drift reverses as the specimen cools: (a) Inclusion is initially faceted as it is heated but becomes round (b) at high temperature.

sponding to the bounding polygon. We defined an appropriate cost function to integrate these constraints and optimized it with dynamic programming. We have implemented a multigrid implementation of this algorithm for high processing speed and higher tolerance of large motion.

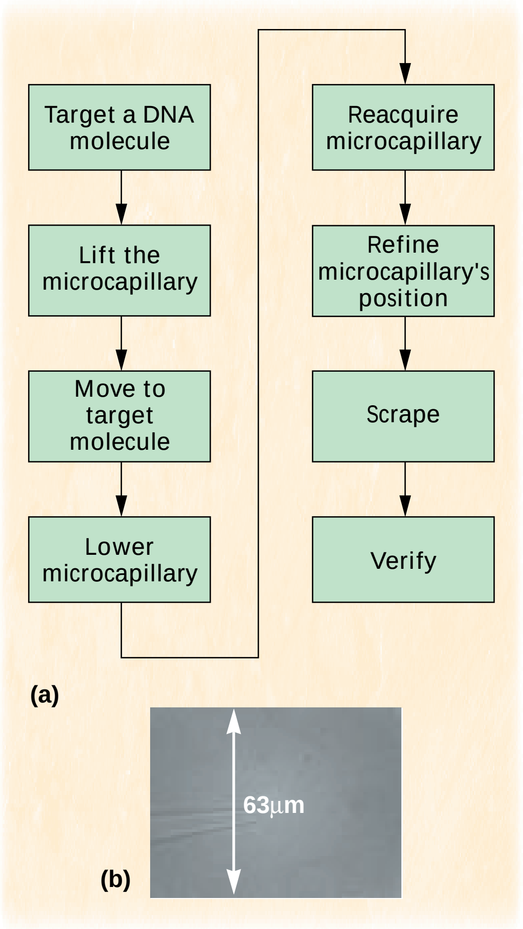

Figure 5. (a) Protocol for scraping a DNA molecule and (b) tip of the microcapillary viewed under a transmitted light source.

The system automatically tracks the object, controls drift, and hides network latencies from the remote user. The drift control is based on tracking and compensating for the contour's centroid. Figure 4 shows the centroid with a crosshair on the image of the reconstructed wavelet coefficients. An arrow indicates the direction of the motion. In addition to topological shape changes that occur during heating and cooling, thermal drift reverses its direction with changes in the temperature gradient's polarity.

Servo-loop control

There are two aspects to servo-loop control: positioning the robotic arm for scraping DNA molecules and correcting thermal drift during in-situ experiments. Both of these applications use deformable contours to acquire, refine, and reposition the mechanical assembly. However, the placement of the robotic arm is imprecise and needs to be more accurate. This is accomplished by moving the microcapillary near the scraping site, reacquiring the microcapillary's position using the deformable contour, repositioning it to where it should be, scraping the site, and then verifying the scraped site by simple thresholding. Figure 5 shows this protocol, and Figure 6 shows an example of microdissection.

In contrast, thermal drift correction is continuous, smooth, and linear. We use a Kalman filter model to predict motion, similar to research reported earlier.[8] Our implementation uses position and velocity to represent the object's internal state. In this context, the model predicts the inclusion's trajectory. As a result, instead of making incremental corrections to the *xy* stage platform, we run the controller at a constant speed and in a direction opposite that of the thermal drift. The speed is then refined at the tracker sampling interval. We discussed details of the Kalman filtering model elsewhere.[4]

SOFTWARE ARCHITECTURE

We use four servers—video, motion, instrument, and data acquisition—in a distributed client-server model. In the architecture shown in Figure 2, each server can interact with any other. The video server captures images and transfers them to the motion server, which manages visual routines and servoing. Using a threaded programming paradigm for parallel decomposition, the visual-routines and servoing module execute asynchronously. The instrument server provides the abstract interface to instrument control for various clients. The data acquisition server provides the low-level interface to underlying hardware.

The motion server's software implementation is interesting. As Figure 7 shows, this server has four asynchronous threads. The instrument thread handles interaction with the instrument server, and we have isolated it for modularity—most PC interactions average about 5 ms. The tracking thread operates at 8 to 10 Hz (depending on the inclusion size) and uses two CPUs for concurrent processing. The compression thread runs at 4 Hz and shares a thread-safe buffer—through locks and conditional wait—with the tracking thread. The focus thread runs at lower throughput and on a single thread over the target region.

Our framework's main benefit is the increased use of a sophisticated instrument (the electron microscope) that had restricted access up to this time because of its sensitive components and requirement for a skilled operator. Secondary benefits include lower costs for conducting individual experiments and increased capability for collaboration among experimenters. These benefits are realized through real-time visual servoing that serves as a computational layer to hide the latency in the wide area network. For more information see http://www-itg.lbl.gov. ❖

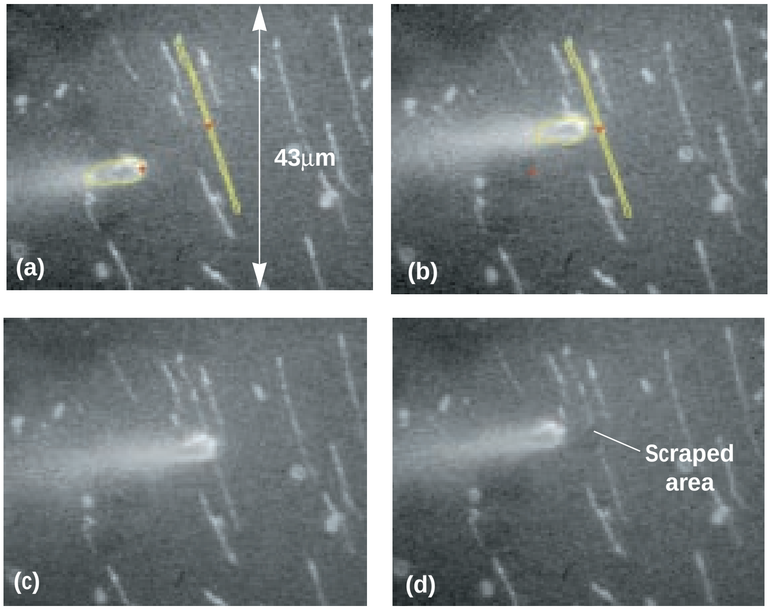

Figure 6. Microdissection of a single DNA molecule under florescence light: (a) original image with microcapillary and several DNA molecules, (b) microcapillary is positioned near the scraping site, (c) microcapillary scrapes the molecule, and (d) a last visual image verifies the scraped area.

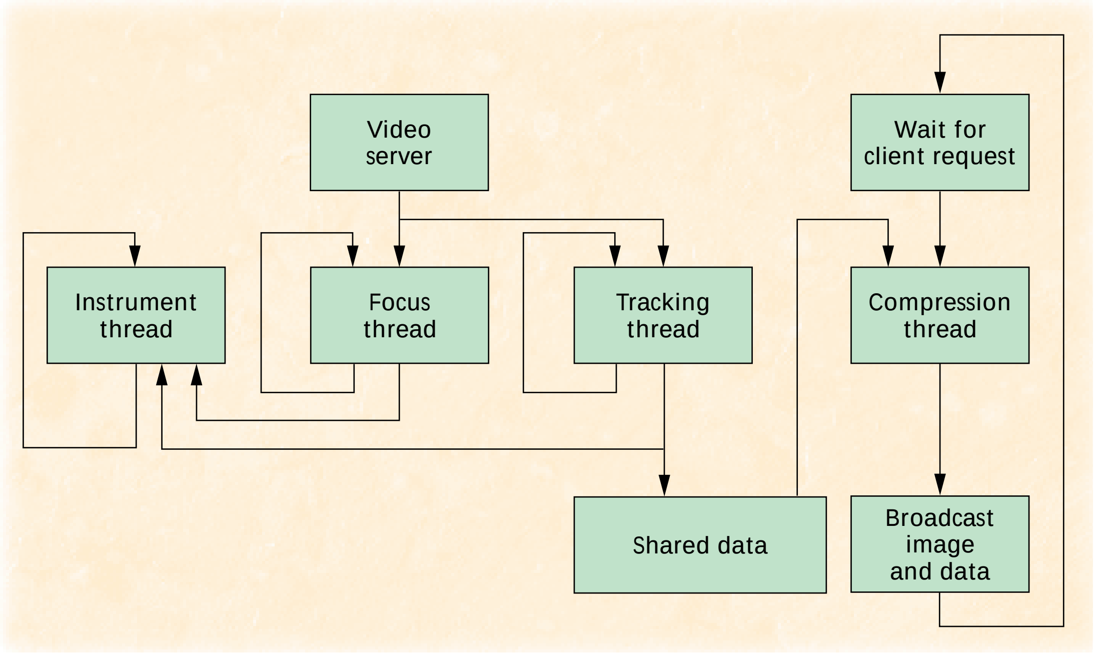

Figure 7. Interaction of various threads during real-time control.

Acknowledgments

We thank the many individuals who have participated in this project, in particular Brian Crowley, Lili Wu, Jin Guojun, Marcos Maestre, Damir Sudar, Mike O'Keefe, Ken Westmacott, and Doug Owen.

This work was supported by the Office of Energy Research; the Office of Computation and Technology Research, Mathematical, Information, and Computational Sciences Division; and the Office of Basic Energy Sciences of the US Department of Energy (under contract No. DE-AC03-76SF00098 with the University of California). The Lawrence Berkeley National Laboratory publication number for this document is LBNL-40078.

References

1. S.J. Young et al., "Implementing a Collaboratory For Microscopic Digital Anatomy," *Int'l J. Supercomputer Applications and High-Performance Computing*, Summer-Fall 1996, pp. 170-181.
2. C. Potter et al., "EVAC: A Virtual Environment for Control of Remote Imaging Instrumentation," *IEEE Computer Graphics and Applications*, July 1996, pp. 62-66.
3. G. Kondraske et al., "Network-Based Infrastructure for Distributed Remote Operations and Robotic Research," *IEEE Trans. Robotics and Automation*, Oct. 1993, pp. 702-704.
4. B. Parvin et al., "Telepresence For In-Situ Microscopy," *Proc. Int'l Conf. Multimedia Computing and Systems*, IEEE CS Press, Los Alamitos, Calif., 1996, pp. 481-488.
5. B. Parvin et al., "Tracking of Tubular Molecules For Scientific Applications," *IEEE Trans. Pattern Analysis and Machine Intelligence*, Aug. 1995, pp. 800-805.
6. B. Parvin, S. Viswanathan, and U. Dahmen, "Tracking of Convex Objects," *Proc. Int'l Symp. Computer Vision*, 1995.
7. J. Bergen et al., "Hierarchical Model-Based Motion Estimation," *Proc. European Conf. Computer Vision*, 1992, pp. 236-252.
8. T.J. Broida and R. Chellappa, "Estimation of Object Motion Parameters From Noisy Images," *IEEE Trans. Pattern Analysis and Machine Intelligence*, 1986, pp. 90-99.
9. I. Daubechies, *Ten Lectures on Wavelets*, SIAM, Philadelphia, 1992.

Bahram Parvin is a staff member in the Information and Computing Sciences Division at the Lawrence Berkeley National Laboratory. He performs research in computer vision, intelligent system design, and collaborative computing. Parvin received his PhD in electrical engineering from the University of Southern California. He holds patents on algorithmic decomposition for a VLSI architecture and intelligent instrument control. He has served on the program committee for several IEEE conferences and is a member of the IEEE.

John R. Taylor is a graduate research assistant at LBNL. His research interests include computer vision, image processing, and distributed multimedia. Taylor received a BA in philosophy from the University of Massachusetts, Amherst, and an MS in computer science from San Francisco State University.

Daniel E. Callahan is a scientist in LBNL's Life Sciences Division. His research interests include spectroscopic studies of DNA conformation and the 3D organization of chromosomes. Callahan received a PhD in chemistry from the University of California, Santa Barbara, and was a postdoctoral fellow in the Division of Biophysics at Johns Hopkins University. He is a member of the Biophysical Society.

William E. Johnston is a senior scientist in LBNL's Information and Computing Sciences Division. His professional interests include high-speed distributed computing, distributed collaboratory environments, distributed security architectures, and clusters of multiprocessor systems in the scientific environment. Johnston received an MA in mathematics from San Francisco State University.

Ulrich Dahmen heads the National Center for Electron Microscopy at the LBNL. He also directs a research program on the crystallography of microstructures, which focuses on the atomic-scale structure of interfaces in materials. Dahmen received his PhD in materials science from the University of California, Berkeley.

Contact Parvin at the Information and Distributed Computing Division, Lawrence Berkeley National Laboratory, Berkeley, CA 94720; parvin@george.lbl.gov.